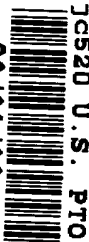


A

08/31/99



Jc520 U.S. PTO

# UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37  
CFR 1.53(b))

Attorney Docket No. C0441/7150 (PDS/TAH)

First Named Inventor or Application Identifier

Dobbins et al.

Express Mail Label No. EL056833040US

Date of Deposit August 31, 1999

Jc542 U.S. PTO

09/30/99



08/31/99

## APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents

## ADDRESS

TO:

Assistant Commissioner for Patents  
Box Patent Application  
Washington, DC 20231

1. ☒ Fee Transmittal Form  
(Submit an original, and a duplicate for fee processing)
2. ☒ Specification [Total pages 26]  
14 pages specification  
1 pages abstract  
11 pages claims  
53 claims
3. ☒ Drawing(s) (35 USC 113) [Total sheets 11]  
☐ Informal ☒ Formal Total drawings 13]
4. ☒ Oath or Declaration [Total pages ]  
a. ☐ Newly executed (original or copy)  
b. ☒ Copy from a prior application (37 CFR 1.63(d))  
(for continuation/divisional with Box 17 completed)  
[Note Box 5 below]  
i. ☐ DELETION OF INVENTOR(S)  
Signed statement attached deleting  
inventor(s) named in the prior application, see  
37 CFR 1.63(d)(2) and 1.33(b).
5. ☐ Incorporation by Reference  
(usable if Box 4b is checked)  
The entire disclosure of the prior application,  
from which a copy of the oath or declaration is  
supplied under Box 4b, is considered as being  
part of the disclosure of the accompanying  
application and is hereby incorporated by  
reference therein.

6. ☐ Microfiche Computer Program (Appendix)
7. ☐ Nucleotide and/or Amino Acid Sequence  
Submission (if applicable, all necessary)  
a. ☐ Computer Readable Copy  
b. ☐ Paper Copy (identical to computer copy)  
c. ☐ Statement verifying identity of above copies

## ACCOMPANYING APPLICATION PARTS

8. ☐ Assignment Papers (cover sheet & documents(s))
9. ☐ 37 CFR 3.73(b) Statement ☒ Power of Attorney  
(when there is an assignee) **COPY**
10. ☐ English Translation of Document (if applicable)
11. ☒ Information Disclosure ☐ Copies of IDS  
Statement (IDS)/PTO-1449 Citations
12. ☐ Preliminary Amendment
13. ☒ Return Receipt Postcard (MPEP 503)  
(Should be specifically itemized)
14. ☐ Small Entity ☐ Statement filed in prior  
Statement(s) application, Status still proper  
and desired
15. ☐ Certified Copy of Priority Document(s)  
(if foreign priority is claimed)

16. Other:

17. If a **CONTINUING APPLICATION**, check appropriate box and supply the requisite information:

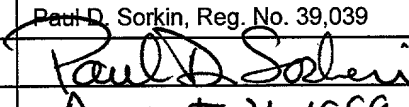
☒ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No.: 08/960,919 filed October 30, 1997 which issued as U.S. Patent 5,946,308 on August 31, 1999; which is a continuation of application no.: 08/559,738 filed November 15, 1995 and which issued as U.S. Patent 5,684,800 on November 4, 1997.

☒ Cancel in this application original claims 1-17 of the prior application before calculating the filing fee.

☒ Amend the specification by inserting before the first line the sentence:

This application is a ☒ continuation ☐ divisional of application serial no. of prior application No.: 08/960,919 which issued as U.S. Patent 5,946,308 on August 31, 1999; which is a continuation of application no.: 08/559,738 filed November 15, 1995 and which issued as U.S. Patent 5,684,800 on November 4, 1997.

18. CORRESPONDENCE ADDRESS					
Correspondence address below					
ATTORNEY'S NAME	Therese A. Hendricks, Reg. No. 30,389				
NAME	Wolf, Greenfield & Sacks, P.C.				
ADDRESS	600 Atlantic Avenue				
CITY	Boston	STATE	MA	ZIP	02210
COUNTRY	USA	TELEPHONE	(617) 720-3500	FAX	(617) 720-2441

19. SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT REQUIRED	
NAME	Paul D. Sorkin, Reg. No. 39,039
SIGNATURE	
DATE	August 31, 1999

Inventor or Identifier: Dobbins et al.

Serial No: Not yet assigned

Filed: Herewith

CHECK BOX, if applicable:

For: Method for Establishing Restricted Broadcast Groups in a Switched Network

☐ **DUPLICATE**

## Fee Calculation Sheet

CLAIMS	FOR	NUMBER FILED	NUMBER EXTRA	RATE	FEE
	TOTAL CLAIMS (37 CFR 1.16(c))	36-20=	16 x	\$18	= \$288.00
	INDEPENDENT CLAIMS (37 CFR 1.16(b))	3-3=	0 x	\$78	= \$0.00
	MULTIPLE DEPENDENT CLAIMS (if applicable) (37 CFR 1.16(d)) +			\$	= \$0.00
				BASIC FEE (37 CFR 1.16(a))	\$ 760.00
				Total of above Calculations =	\$1048.00
				Reduction by 50% for filing by small entity (Note 37 CFR 1.9, 1.27, 1.28).	\$1048.00
				Assignment Recordation Fee (if any)	\$ 0.00
				Other Fees (e.g., Petition for Extension of Time), if any <b>NOTE: Enter small-entity amount if applicable.</b>	\$ 0.00
				TOTAL =	\$1048.00

1. A check in the amount of \$1048.00 is enclosed.

## General Authorization to Charge Deposit Account and General Request for Extension of Time

2. a. ☒ If the filing of any paper in this application necessitates the payment of a fee under 37 CFR ☒ 1.16 ☒ 1.17 or ☐ 1.18, and the fee due is in an amount different from any enclosed check or if no check is enclosed, the Commissioner is hereby authorized to charge any deficiency or credit any overpayment to Deposit Account No. 23/2825.
- b. ☒ The applicant hereby revokes any prior authorization to charge a fee due under 37 CFR ☐ 1.16 ☐ 1.17 or ☒ 1.18.
3. If the filing of any paper in this application necessitates an extension of time under 37 CFR ☒ 1.136(a), the applicant hereby requests such extension of time. If the fee due is in an amount different from any enclosed check or if no check is enclosed, the Commissioner is hereby authorized to charge any deficiency or credit any overpayment to Deposit Account No. 23/2825.



Paul D. Sorkin, Reg. No. 39,039  
 Wolf, Greenfield & Sacks, P.C.  
 600 Atlantic Avenue  
 Boston, MA 02210-2211  
 (617) 720-3500  
 Attorneys of Record

Express Mail Label: ELO56833040US  
Date of Deposit: August 31, 1999

## **METHOD FOR ESTABLISHING RESTRICTED BROADCAST GROUPS IN A SWITCHED NETWORK**

### **Field of the Invention**

5 This invention relates to packet switched data communications networks, and more particularly to an apparatus and method for establishing restricted broadcast groups known as virtual LANs (VLANs) which provide a simple but robust mechanism that allows broadcast and/or multicast packets to be "flooded" through a switched domain and transmitted only to those users or ports defined for a particular VLAN.

### **Related Applications**

10 The subject matter of the above application may be advantageously combined with the subject matters of the following copending and commonly owned applications, which are hereby incorporated by reference in their entirety:

- 15 • U.S. Serial No. 08/188,238 entitled "Network Having Secure Fast Packet Switching And Guaranteed Quality Of Service," filed January 28, 1994 by Kurt Dobbins et al.; and
- U.S. Serial No. 08/187,856 entitled "Distributed Chassis Agent For Network Management," filed January 28, 1994 by Brendan Fee et al.

### **Background of the Invention**

20 Most data communications networks today rely heavily on shared medium, packet-based LAN technologies for both access and backbone connections. These networks use bridges and routers to connect multiple LANs into global internets. An internet router must be capable of  
25 processing packets based on many different protocols, such as IP, IPX, DECNET, AppleTALK, OSI, SNA and others. The complexities of building networks capable of routing packets on the global internet using different protocols is a challenge for both vendors and users.

30 In U.S. Serial No. 08/188,238 to Dobbins (see related applications above), there is described a new secure fast packet switching (SFPS) technology which provides the same or better reliability and security as routers, but with much greater performance and without an increase in cost. The SFPS system avoids the complexities and costs of providing multi-protocol routers. Also, the SFPS system provides capabilities which routers do not, such as the ability to create separate logical work group LANs on the same physical network and the ability to

guarantee a quality of service (QOS) by providing dedicated switched paths through the network.

SFPS provides high performance packet switching based on physical layer addresses such as source and destination MAC IDs -- the unique medium access control (MAC) address assigned to each end system by the IEEE. End-to-end connections are determined by a network management application that provides security and best path routing determinations based on a number of constraints. By switching packets based only on MAC layer information, network infrastructure remains protocol insensitive.

More specifically, SFPS uses source and destination MAC addresses which alone, or in combination with an input port on a switch, form a unique "connection identifier" for any communication exchange between designated end systems. As an example:

input port = 2

source MAC address = 00:00:1D:01:02:03

destination MAC address = 00:00:1D:11:22:33

together form a "tuple" bound to a specific uni-directional flow from a source address to a destination address. All packets that have this tuple are automatically switched according to the operation of the SFPS.

Network infrastructures are built around a core switching fabric, which provides the physical paths or routes that allow users to send information to one another. Access to the switching fabric is gained through an access port. Access ports provide several functions - most importantly, they provide security and accounting services. End systems such as personal computers (PCs), workstations and servers connect to an access port using one or more access technologies such as Ethernet, Token Ring, FDDI, or ATM.

In traditional bridge and router devices, each packet is treated as an independent unit of data which is individually processed by application of access and security constraints, as well as path determination. In contrast, with SFPS this processing is done only on initial probe packets which are decoded, and through use of a central directory of end system constraints policy, call attributes, location, paths, quality of service, etc., the connection is either rejected or accepted. If accepted, the path is determined and switches along the path are "programmed" to allow subsequent packets on this "connection" to be switched. In either case, subsequent datagrams are either switched or discarded without having to reapply all of the security and access control and path determination logic.

The SFPS switching technology may be constructed as: software objects which exist in embedded devices as firmware; software objects which are part of an application on a commercial computer system; application specific integrated circuits (ASIC); or functionally equivalent hardware components.

5 It is common for internetworking devices to "route" the protocols that a device supports, and "bridge" the protocols that are not supported for routing. In addition, some protocol frames (such as DECs LAT) are actually unroutable. In SFPS switches, there are protocol-specific call processors to route protocol-specific broadcast frames (note that unicast frames can be processed by a "generic" call processor  
10 that does not decode or translate the frame at all, but instead makes the connection request based on the source and destination unicast MAC addresses in the frame). However, a problem arises in that an SFPS switch has nothing equivalent to bridging of multicast and broadcast packets for non-supported protocols. Thus, until a protocol-specific call processor is implemented in a switch, the switch must discard any broadcast or  
15 multicast frames it does not understand.

#### Summary of the Invention

A method and apparatus are provided for establishing restricted broadcast groups within a switching fabric, known as virtual LANs (VLANs). The VLANs provide a simple but robust  
20 mechanism for allowing broadcast and multicast packets to be "flooded" through the switching fabric and transmitted only to those users or ports defined for a particular VLAN.

More specifically, the switched network includes a plurality of end systems and switches connected by links. The switches have access ports connected to end systems and network ports connected to other switches. Each end system has a unique physical layer address, e.g., MAC  
25 address. In accordance with this invention, different virtual LAN identifiers (known as VLAN-IDs) are assigned to different subsets of associated end systems or access ports. Each access switch maintains a first table for mapping VLAN-IDs to associated end systems and/or access ports (the End System/VLAN table). Each access switch also maintains a second table for mapping access ports (of associated end systems) to associated VLAN-IDs (the VLAN/Access  
30 Port table).

According to a first embodiment, the restricted V-LAN flooding is used for broadcast packets of a protocol not supported by the switches. When an original broadcast packet (of an unsupported protocol) is received by a first access switch from a first end system, the first switch

determines and adds a VLAN header to the original data packet to create a VLAN packet. The VLAN header includes designated VLAN-IDs from the first table. The designated VLAN-IDs are assigned based on the physical source address of the first end system. The first access switch then forwards the VLAN packet to all other switches on a multicast channel of point-to-point connections between the switches. The first switch also forwards the original broadcast packet out the access ports identified in the second table for the designated VLANs (except the originating port).

The other switches receive the VLAN packet and extract the designated VLAN-IDs from the VLAN header and then forward the original packet out the access ports, defined in the other switch's second table, for the designated VLAN-IDs.

Other embodiments include the designation of a default VLAN-ID which maps to all access ports, a reserved VLAN-ID for use with multicast packets, and restricted flooding for packets directed to an undiscovered end system. Still another embodiment provides a single or distributed network server on the multicast channel (between switches) to handle broadcast and multicast packets, which embodiment scales better for larger networks.

More specific methods and a particular apparatus for implementing the present invention are described in the following detailed description and drawings.

#### Brief Description of the Drawings

Fig. 1 is a schematic illustration of a network topology built with SFPS switches.

Fig. 2 is a schematic illustration of an SFPS switch.

Fig. 3 is a logical view of an SFPS switch.

Fig. 4, combining Figs. 4-A and 4-B, is a flowchart showing processing of a data packet by an SFPS switch.

Fig. 5 is a schematic illustration of a network topology including three virtual networks (VLAN 100, VLAN 5, and VLAN 20) according to the present invention.

Fig. 6 shows an end system table for mapping VLAN-IDs to associated end systems.

Fig. 7 shows a port table for mapping access ports (of associated end systems) to associated VLAN-IDs.

Fig. 8 shows one embodiment of a VLAN packet, in which a VLAN header is appended to an original data packet.

Fig. 9 is a schematic illustration of a network topology utilizing a default VLAN according to the present invention.

Fig. 10 is a schematic illustration of a network topology utilizing a distributed VBUS server according to an alternative embodiment of the present invention.

Fig. 11 is a flow chart showing the redirected flow of a broadcast or multicast packet to the VBUS service.

Fig. 12 is a flow chart showing the call processing performed by the VBUS service.

Fig. 13 is a flow chart showing the channel listening process of the VBUS service.

### Detailed Description

#### The SFPS Network -- Switching of Unicast Packet With Generic Call Processor and Switching Of Protocol-Supported Broadcast Packets

According to one embodiment, the establishment of VLANs for transmitting broadcast/multicast packets of non-supported protocols is intended for use in the SFPS switched network described in U.S. Serial no. 08/188,238. The following is a general description of the operation of switching "unicast" packets on that network, as illustrated in Figs. 1-4.

Fig. 1 shows a representative network topology built with six secure fast packet switches (SFPS) labeled S1-S6 connected by links L. Each switch has for example four ports; some ports are labeled A for access and some are labeled N for network. The end systems are connected to the access ports by links L and are labeled "M\_". One end system is a network management station or server (NMS) M10, which includes a connection server.

Fig. 2 is a schematic illustration of an SFPS switch 91 having a plurality of ports 92. A host port 93 connects the switch to its host CPU 90, which may be an i960 microprocessor sold by Intel Corporation. The host CPU is connected to a system management bus (SMB) for receipt and transmission of discovery and other control messages.

Fig. 3 illustrates the internal operation of the switch. The SFPS switch 86 includes in ports 80, out ports 81, a connection database 82, a look-up engine 83, and a multilevel programmable arbiter MPA 84. The switch 86 sends and receives messages from the host agent 85, which includes a management agent 87, a discovery agent 88, and a call processing agent 89.

The management agent 87 provides external control of the configuration and operation of the SFPS switch, through the network management system.

The discovery agent 88 provides a mapping of end systems to switching ports through a



passive listening (snooping) capability and a registering of end system addresses and port locations of the host switch with a common external directory. Adjacent switches are also discovered and mapped, but this may be done with an explicit switch-to-switch protocol (nonpassive).

5           The call processor 89 provides a means for requesting connections to be established between two end systems. Unicast frames are handled by a "generic" call processor which programs the switches based on the source and destination MAC addresses. In a case where the source and destination MAC addresses are not in the packet frame, i.e., usually in a frame that has a broadcast -- all hosts -- MAC address, a "protocol-specific" call processor (if available) will decode the packet to find source or destination network addresses and use these to map back to the physical addresses via the external directory. Once the end system MAC addresses are known, the protocol-specific call processor will then request the connection between the end systems. If the broadcast frame was a probe or address resolution packet (i.e., an implied connection request), the call processor will return a probe reply as a "proxy" which gives the destination end system MAC addresses. Subsequently, the source end system can then send packets directly to the destination based on its MAC address.

Fig. 4 is a flow chart illustrating what happens from the time a data packet is received on an in port of the switch, until it is sent on the correct out port.

Referring to Fig. 4-A, in step 300 the host CPU 90 is initialized. The host programs the connection database 82 to send any "unknown" or "broadcast" connections to the host port (step 301). The switch then waits for a packet to arrive (step 302). Once a packet has arrived (step 303), the switch extracts the source MAC address, destination MAC address, and identifies the inbound port on which the packet was received (step 304). The look-up engine 83 checks to see whether this source-destination pair is already located in the connection database 82 (step 305). If it is not found, the packet is given to the host agent 85 (step 308). The call processor and the host agent determine whether it is a broadcast destination (step 309). If the answer is yes, a protocol-specific call processor (if available) decodes the packet to find the network protocol source and destination addresses (steps 310-311). A different protocol decode logic would be provided for each network protocol. For example, in the IP protocol, if an ARP request is received, the call processor would get the target IP address (step 312). It would then ask the external directory for the MAC address of the destination IP (step 313). In the next step 314, the directory sends the MAC destination address back to the call processor. The call processor 89 then asks the connection server (SCS) to set up a connection between the source MAC and

destination MAC (step 315). The call processor 89 forms an ARP reply packet by putting the destination MAC address inside the packet (step 316), and the call processor sends a reply to the source address (step 317). It should be noted that this reply allows the source end system to update its private mapping of the destination IP address to a nonbroadcast MAC address. All subsequent packets to this destination IP address will be properly framed with the source and destination MAC address for which connections will now exist.

Note that if no call processor exists which supports the relevant protocol, the broadcast packet is dropped (step 321). The present invention is a method of handling such packets.

If the packet is not a broadcast packet, it is given to the "generic" call processor (treated as an unknown SA-DA connection -- step 318), which asks the connection server to set up a connection and forward the packet (step 319); the call processor then discards the packet (step 320).

Returning to step 305, if the source and destination MAC pair are found in the connection database 82, the data packet is sent to the switch output(s) 81 defined in the database (step 306). In next step 307, the management agent 87 collects statistics regarding transmissions through the switch and sends them to the SCS (connection server).

#### Restricted Broadcast Groups For Non-Supported Broadcast, MultiCast and Unknown Unicast Packets

Fig. 5 illustrates generally a logical view of the present invention for establishing restricted broadcast groups or virtual LANs (VLANs) within a switched network. The representative network 10 has four SFPS switches 11-14, all of the switches being connected by physical links forming point-to-point connections 15, and which physical connections together form a logical multicast channel 16. The multicast channel 16 connects the network ports of all switches. A plurality of end systems 20A-20L extend from access ports on the various switches 11-14. The end systems are shown grouped into different subsets known as virtual LANs 17, 18 and 19, which are given VLAN identifiers VLAN 100, VLAN 5, and VLAN 20, respectively. As shown in Fig. 5, "VLAN 20" includes end systems 20B, 20C, 20J and 20K. "VLAN 5" includes end systems 20D, 20G, 20H and 20L. "VLAN 100" includes end systems 20A, 20B, 20D, 20F, 20H and 20I.

During a discovery time, as each switch 11-14 is discovered, it is put in a point-to-point connection that connects all SFPS switches. This forms the multicast channel 16 which all

switches use between themselves.

Also during the discovery time, each switch 11-14 discovers its associated end systems (i.e., switch 11 discovers end systems 20A, 20B, 20C) and enters these end systems in a common directory which assigns VLAN-IDs to the end systems. The directory returns a mapping of  
5 VLAN-IDs and associated end systems, which mapping each switch uses to build two internal tables: a first table that lists the VLAN-ID for each end system (the End System/  
VLAN Table -- see Fig. 6), and a second table that defines a port mask for each VLAN-ID (the  
VLAN/Access Port Table -- see Fig. 7).

During real time operation of the system, a first switch (for example switch 11) receives a  
10 broadcast or multicast packet that it cannot process with a protocol-specific call processor. The switch will encapsulate the original packet and insert a VLAN header containing a list of  
VLAN-IDs for the source end system (see Fig. 8), before flooding the encapsulated (VLAN)  
packet out the multicast channel 16 to all other switches. For example, if first switch 11 receives  
a broadcast packet from first end system 20B, switch 11 returns from its end system table (Fig 6)  
15 that VLAN 100 and VLAN 20 are associated with source end system 20B. First switch 11 will  
insert VLAN 100 and VLAN 20 into the VLAN header (Fig. 8). In addition, first switch 11  
determines the port masks for VLAN 100 and VLAN 20 from its port table (Fig. 7), and then  
sends the original broadcast packet out all access ports of the first switch in VLAN 100 or VLAN  
20 (except for the source port 2); in this case, the original packet is sent out access port 1, which  
connects to end system 20A, also in VLAN 100, and out access port 3, which connects to end  
system 20C, also in VLAN 20.

As each of switches 12, 13 and 14 receive the VLAN packet on multicast channel 16, they  
strip off the encapsulated VLAN header and look up in their respective VLAN/Access Port table  
for any associate mapping to VLAN 100 and VLAN 20. Switch 12 determines in its port table  
25 that it has associated access ports to end systems 20D and 20F designated for VLAN 100.  
Similarly, switch 13 determines from its port table that it has associated access ports to end  
systems 20H and 20I for VLAN 100. Switch 14 determines from its port table that it has  
associated access ports to end systems 20J and 20K for VLAN 20. The original packet is thus  
transmitted out the access ports to end systems 20D, 20F, 20H, 20I, 20J and 20K.

30 The following describes the changes and additional functionality required of the SFPS  
access switches to support the establishment of VLANs for multicast and broadcast packets.  
Switches with only network ports continue to function as described in prior U.S. Serial No.  
08/188,238 to Dobbins et al.

### The Switch-To-Switch Multicast Channel

Each SFPS switch supports the multicast channel 16 by having a connection in each switch that connects it to all other switches in the network (or within a subsection of the network, such as a domain). This is in essence a point-to-

5 multipoint connection in each switch. It should be noted that this multipoint connection is only between the switches themselves, which scales better than having a multipoint connection between all users (end systems).

A connection server (i.e., M10 in Fig. 1), which includes a common directory of all switches, has the responsibility to program the multicast channel connection each time a  
10 new switch joins or leaves the topology, i.e., such a change may be detected by neighbor advertisement signals sent by the switches.

### The End System/VLAN Table (Fig. 6)

Each switch that has an access port maintains a table of end systems heard on each  
15 access port, and a list of VLANs to which each end system belongs. An end system can belong to more than one VLAN at any given time.

The assignment of VLAN-IDs may be accomplished in several ways. First, the VLAN-IDs may be maintained by a common directory. For example, as each end system is discovered by an access switch, it is registered with a common directory of end systems  
20 for the entire network, and the directory then returns a list of VLAN-IDs to the access switch with the "End System Discovery Message ACK." Alternatively, a management application may administratively assign the VLAN-IDs, and manage the end system and port tables in the switch. As a further alternative, an access switch may send a Resolve signal to a directory, which directory then returns a mapping of VLAN-IDs for an  
25 associated end system.

### The VLAN/Access Port Table (Fig. 7)

Each switch having an access port maintains a port table which maps VLANs to associated access ports. This table may be filled in dynamically through the implicit  
30 mapping of VLANs to end systems. Each time a VLAN is mapped to an end system, it is automatically inserted in a port-mapping entry for the source port on which the end system is located. Ports, like end systems, can belong to more than one VLAN at any given time; the port's VLAN mapping strictly

depends upon the VLAN of the end systems on it. The out ports for each VLAN entry in the table essentially define the flood mask for the access ports.

### The Default VLAN-ID

5        A default VLAN-ID (VLAN-ID = 1) may be used to map an end system to a VLAN when no VLAN-ID had been administratively assigned. The default VLAN is a special case which maps to all access ports. This allows flooding out all ports when no VLAN is defined. For example, Fig. 9 illustrates a network of four switches 30, 31, 32, 33, connected by multicast channel 35, prior to assignment of  
10      any specific VLAN-IDs such that all end systems fall within a default VLAN 36.

### The VLAN Call Processor

15      The VLAN call processor is essentially a default call processor for broadcast/multicast packets for which no protocol-specific call processor exists. For example, an ARP call processor would be able to decode an ARP broadcast message.

20      The VLAN call processor would take any packet it receives and then encapsulate the broadcast/multicast packet in a header, the header containing a list of VLAN-IDs on which the packet belongs. The VLAN-ID list may be determined by using the source MAC address of the original packet and doing a look-up in the end system table. In this example, VLAN-IDs are determined based on the source rather than the destination. Once this encapsulated packet is formed, it is then forwarded on the multicast channel to all other switches. The original packet would also be given to the local forwarder.

### The Local Forwarder

25      Each switch has a process that listens on the multicast channel 16. This process is responsible for processing any encapsulated frames (VLAN packets) sent from other switches. When the VLAN packet is received, it is stripped of its VLAN-ID list in the header. For every entry in the VLAN-ID list, the local port table is searched for a matching entry. The local forwarder then forwards the original data packet out any ports  
30      that are mapped to the VLAN-ID. If the VLAN-ID is the default VLAN-ID (= 1), then the original packet is flooded out all access ports on the switch. If no VLAN-IDs match, then the packet is discarded.

### The Central Connection Server and Common Directory

A central connection server programs the point-to-multipoint connections between all of the SFPS switches, as there is no provision in each switch to do so (see M10 in Fig.

1). Thus, any time the connection server "discovers" a change in a switched topology, it has to reprogram the multicast channel between the switches.

The server accesses a common directory for mapping end systems to VLAN-IDs. A management application may provide this on the front end, and in addition provide for changes to the mapping in the directory itself and in any switches that have been informed of the mapping. Any end system not defined with a VLAN would default to VLAN-1.

If the VLAN assignment is done inside an End System Discovery Message ACK, then a new TLV list is added to the message. This functions similar to an "alias" field in which more than one are allowed since multiple VLANs could be returned. If the VLAN assignment is done with Resolve messages, then only a new TAG type has to be assigned since the message supports returning multiple resolutions. The semantics would be "resolve this end system to its VLAN-IDs." If the assignment was done completely out of band, then no signalling changes would be required.

### Reserved VLAN-IDs

In the previous embodiment, broadcast and multicast packets are propagated through the switches based on the VLAN-IDs to which the source belongs. In some cases, mostly with multicast frames, it may be desirable to map a VLAN to the destination, e.g., OSPF packets.

This may be accomplished by allowing the switches to support "well-known" VLANs without any run-time assignment. If a switch receives a packet destined for a reserved VLAN, it would encapsulate it and set the VLAN list without mapping it to the end system table. The packet would then be forwarded out the multicast channel and any switches supporting the reserved VLAN (or having heard a reserved VLAN-type packet), would flood the original packet out.

### Unicast Flooding

VLANs may be supported for unicast frames, for example if a call processor has not yet discovered the end system. This works similar to the broadcast/multicast operation

except that instead of mapping the outports at each flooding switch, each switch would look up the destination unicast address in the end system table and send the original packet out the port on which the end system belongs.

## 5 VBUS Service

Since point-to-point connections between switches does not scale well, in an alternative embodiment each switch has a connection to a single (or distributed) server in the network which will forward broadcast and multicast packets. This service, referred to as the Virtual Broadcast/Unknown Service (VBUS), is distributed into all SFPS switches  
10 in a first implementation as illustrated in Fig. 10. Switches 61, 62, 63, 64 are connected by multicast channel 66, and each switch includes the distributed VBUS service 65.

Fig. 11 illustrates the redirected flow of data packets for the VBUS service. When a first switch receives a broadcast or multicast packet (step 40), it first determines whether the packet was received on an access port (step 41). If no, the packet is discarded  
15 (step 47). If yes, the packet is passed to a redirector queue (step 42), and if a call processor supports the packet type (step 43), the redirector delivers the packet to the protocol-specific call processor (step 44). If not, the packet is passed to the VBUS call processor (step 45). The redirector queue then handles the next packet on the queue (step 46).

Fig. 12 illustrates the operation of the VBUS call processor. Each switch  
20 listens for source addresses heard on each access port (step 48). The call processor then updates the End System/VLAN table with the access port and end systems heard (step 49). The call processor then creates a signal entry (step 50) which is sent to the connection server (step 51), which formats a response to the signal (step 52). The connection server returns a signal with the associated VLAN list, which is received by the call processor  
25 (step 53). The call processor gets the associated access ports from the VLAN/Access Port Table (step 54) and sends out the original packet on the associated access ports (step 55). The call processor gets the network ports from the switch's connection table (step 56), encapsulates the packet with the VLAN header (step 57), and sends the encapsulated packet out the network ports to the other switches (step 58). The call processor then  
30 deletes the signal entry (step 59) and returns to start (step 60).

Fig. 13 illustrates the operation of the VBUS channel listener. When a packet is received on a network port (step 70), it first determines whether there is a known connection in the connection database 82 (step 71), and if so, it forwards the packet out the

appropriate output (step 72). If there is no connection, it determines whether this is a VBUS packet (step 73). If no, it returns to the beginning. If it is a VBUS packet, the packet is handed to a VBUS port (step 74) and the VLAN list is extracted from the header (step 75). The access ports are obtained from the VLAN/Access Port Table (step 76), and  
5 the encapsulation header removed from the packet (step 77). The original packet is then sent out the associated access ports defined in the table (step 78).

In one embodiment, the switch provides a MIB interface to allow an external application to assign VLAN-IDs to access ports and/or end systems. The simplest model is to program VLAN-IDs to the switched ports only; under this model, the administration is  
10 simpler and the VLAN assignment to end systems is implied by the switched port to which the end systems are physically connected. A more robust model would map the VLAN-IDs from policy work group definitions.

The application interface may be provided with an SNMP (Simple Network Management Protocol) MIB (management information base) which allows a simple  
15 interface to program connections via a single SNMP set message. The MIB interface provides the following semantics:

*(map, unmap)*[SFPS VLAN-ID][inPort][userMAC]

This verb set assigns (and removes) a user MAC address and switch port to (or from) a specific VLAN.

20 *(map-port, unmap-port)*[SFPS VLAN-ID][inPort]

This verb set assigns (and removes) a switch port to (or from) a specific VLAN. The switches provide managed objects accessible by the MIB which are all accessed with standard SNMP get, get next, and set messages.

In one embodiment, a VLAN status table is provided. This table allows an  
25 entire VLAN to be enable or disabled regardless of the number of user or switch ports assigned to the VLAN in the switch. Thus, it is possible to shut off a particular VLAN inside a particular switch without having to administer each individual switch port or end system.

One goal of the VBUS service is to require minimal support from the  
30 network server. The only server requirement is providing each switch with a connection to all other switches in the network (domain), which in effect provides the multicast channel for flooding VLAN packets.

While there have been shown and described several embodiments of the present



[illegible]

**CLAIMS**

1. A method of forwarding broadcast data packets in a switched data communications network, the network including a plurality of end systems and switches connected by links, the switches having access ports connected to end systems and network ports connected to other switches, and each end system having a unique physical address, the method comprising the steps of:

- a. assigning different virtual LAN identifiers (VLAN-IDs) to different subsets of associated end systems or access ports;
- b. maintaining a first table for mapping the VLAN-IDs to the associated end systems or access ports;
- c. maintaining a second table for mapping the access ports to the associated VLAN-IDs;
- d. when a broadcast packet is received from a source end system at a receiving access port of a first switch:
  - i) reviewing the first table for one or more VLAN-IDs associated with the source end system or receiving access port;
  - ii) encapsulating the packet by adding a header with the associated VLAN-IDs;
  - iii) forwarding the encapsulated packet to all other switches in the network;
  - iv) reviewing the second table for the access ports on the first switch associated with the associated VLAN-IDs and forwarding the broadcast packet out the associated access ports; and
- e. when the encapsulated packet is received at a next switch:
  - i) stripping the header from the encapsulated packet and determining the associated VLAN-IDs;
  - ii) reviewing the second table for the access ports associated with the associated VLAN-IDs; and
  - iii) forwarding the broadcast packet out the associated access ports.

2. The method of claim 1, wherein steps a-c include:  
maintaining a common registry of assigned VLAN-IDs; and

maintaining the first and second tables at each switch.

3. The method of claim 2, wherein steps a-c include:  
registering each end system with the common registry, and returning a list of  
5 assigned VLAN-IDs from the common registry to each switch.

4. The method of claim 2, wherein steps a-c include:  
providing common management of the first and second tables at each switch.

10 5. The method of claim 2, wherein steps a-c include:  
sending a signal from the first switch to the common registry to resolve an  
end system to its assigned VLAN-IDs.

15 6. The method of claim 1, wherein steps a-c include:  
prior to assigning a VLAN-ID to a specific end system, maintaining a default  
VLAN-ID for that specific end system which maps to all access ports.

20 7. The method of claim 1, wherein step d includes:  
maintaining a multicast channel of connections between all switches and  
sending the encapsulated packet on the multicast channel to all other switches in the  
network.

8. The method of claim 7, wherein step d includes:  
maintaining a point-to-multipoint connection from each switch to all other  
switches in the network.

5 9. The method of claim 7, including:  
providing a network server and maintaining a point-to-point connection  
between the server and each switch; and  
forwarding all broadcast packets received at the first switch to the network  
server, the network server performing steps b-d.

10 10. The method of claim 7, wherein each end system is registered with a  
common registry which programs the multi-cast channel.

11. The method of claim 1, wherein steps b-c include:  
15 maintaining the first and second tables at each switch.

12. The method of claim 1, wherein step b includes:  
listening to end systems heard on respective access ports at each switch and  
maintaining the end systems heard and their respective access ports in the first table  
20 at the respective switch.

13. The method of claim 1, further comprising:  
assigning reserved VLAN-IDs without limitation as to end system and access  
port; and  
25 providing at least one switch which encapsulates the broadcast packet by  
adding a header with a reserved VLAN-ID, forwarding the encapsulated packet to all  
other switches, and forwarding the broadcast packet out the access ports on the at least  
one switch.

14. The method of claim 1, wherein step b further includes:

listening to end systems heard on respective access ports at each switch and maintaining the end systems heard and their respective access ports in the first table at the respective switch; and

5 upon receipt of a unicast packet for a destination end system unknown to the first switch, completing steps d.i-iv) and e.i) and then at the next switch reviewing the first table for the respective access port for the destination end system and forwarding the packet out the respective access port.

10 15. The method of claim 1, wherein steps b-c include:

maintaining a Management Information Base (MIB) interface at each switch for programming the first and second tables.

16. The method of claim 15, wherein steps b-c include:

15 using a Simple Network Management Protocol (SNMP) set message for maintaining the first and second tables.

17. The method of claim 1, wherein steps b-c include:

20 maintaining a VLAN status table at each switch for enabling and disabling an entire VLAN-ID.

18. A computer-readable storage medium comprising program instructions for restricting flooding of a data packet, of one of a broadcast, multicast and unknown destination type, in a switched data communications network, the network including a plurality of end systems and switches connected by links, the switches having access ports connected to end systems and network ports connected to other switches, the program instructions causing the network to:

- a. assign at least one identifier to a respective subset of end systems;
- b. map the at least one assigned identifier to an access port attached to at least one end system in the respective subset of end systems; and
- c. when the data packet is received from a source end system at a receiving access port of a first switch:
  - i) determine one or more identifiers associated with the source end system;
  - ii) encapsulate the data packet by adding a header with the one or more determined identifiers;
  - iii) forward the encapsulated data packet to all or a subset of other switches in the network; and
  - iv) determine if at least one access port other than the receiving access port on the first switch is associated with the one or more determined identifiers and forward the data packet out the at least one determined access port.

19. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network to:

- d. when the encapsulated data packet is received at a second switch with access ports:
  - i) strip the header from the encapsulated data packet and to determine the one or more encapsulated identifiers in the header of the encapsulated data packet;
  - ii) determine if at least one access port of the second switch is associated with the one or more encapsulated identifiers; and
  - iii) forward the data packet out the at least one determined access port of the second switch.

20. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network to, if in step c(iv) no other access port is determined, discard the data packet.

21. The computer-readable storage medium as recited in claim 18, further comprising, in step b, instructions to cause the network to:

- maintain a first table in each switch to relate the at least one assigned identifier to the end systems or access ports of the respective switch; and
- maintain a second table in each switch to relate the access ports of the respective switch to assigned identifiers.

22. The computer-readable storage medium as recited in claim 21, further comprising, in step c.i), instructions to cause the network to:

- review the first table for the one or more identifiers associated with the source end system or the receiving access port.

23. The computer-readable storage medium as recited in claim 22, further comprising, in step c.iv), instructions to cause the network to:

- review the second table for an access port associated with the one or more determined identifiers.

24. The computer-readable storage medium as recited in claim 23, wherein the assigned identifier is a virtual LAN identifier.

25. The computer-readable storage medium as recited in claim 18, wherein the received data packet is of a protocol not supported by a protocol-specific call processor in the first switch.

26. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network to:

- maintain a common registry of assigned identifiers.

27. The computer-readable storage medium as recited in claim 26, further comprising instructions to cause the network to:

- register each end system or access port with the common registry, and

return a list of assigned identifiers from the common registry to each switch for the end systems or access ports of the respective switch.

28. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network to:

maintain the mapping at each switch for the end system or access ports of the respective switch.

29. The computer-readable storage medium as recited in claim 26, further comprising instructions to cause the network to:

send a signal from the first switch to the common registry to resolve an end system or access port to its assigned identifiers.

30. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network to:

prior to assigning an identifier to a specific end system or access port, maintain a default identifier for that specific end system or access port which maps to predetermined access ports.

31. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network to:

maintain a multicast channel of connections between all or a subset of switches and wherein step c(iii) comprises forwarding the encapsulated packet on the multicast channel.

32. The computer-readable storage medium as recited in claim 31, wherein the channel includes:

a point-to-multipoint connection from each switch to all other switches in the network.

33. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network, at step c(iii), to provide a distributed service in the switches for forwarding the encapsulated data packet.



34. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network to assign the identifier based on a policy work group definition.

35. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network to:  
maintain at least one mapping table at each switch for performing the mapping step.

36. The computer-readable storage medium as recited in claim 35, further comprising instructions to cause the network to:  
listen to end systems heard on respective access ports at each switch and maintain the end systems heard and their respective access ports in the mapping table at the respective switch.

37. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network to:  
assign a reserved identifier without limitation as to end system and access port.

38. The computer-readable storage medium as recited in claim 19, further comprising instructions to cause the network to:  
listen to end systems heard on respective access ports at each switch and to maintain the end systems heard and their respective access ports in a mapping table at the respective switch; and  
upon receipt of a unicast packet for a destination end system unknown to the first switch, complete step c for the unicast packet and then at the next switch review the mapping table for the respective access port for the destination end system and forward the packet out the respective access port.

39. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network to:  
maintain a Management Information Base (MIB) interface at each switch for programming at least one mapping table, the mapping table being used to perform the mapping step.

40. The computer-readable storage medium as recited in claim 39, further comprising instructions to cause the network to:  
use a Simple Network Management Protocol (SNMP) set message to maintain the mapping table at each switch.

5 41. The computer-readable storage medium as recited in claim 18, further comprising instructions to cause the network to:  
maintain a status table at each switch to enable and disable a respective subset.

42. Computer software, residing on a computer-readable storage medium, comprising instructions for use in a switch in a switched communications network including a plurality of end systems, the computer software for restricting flooding of a data packet selected from the group consisting of a broadcast packet, a multicast packet, and an unknown destination packet of a protocol not supported by a call processor in a switch which receives the data packet, the instructions causing the switch to:

10 assign at least one identifier to a respective subset of end systems;  
15 map the at least one assigned identifier to an access port of the access switch attached to at least one end system in the respective subset of end systems;  
upon receipt of the data packet at the access switch, encapsulate the data packet with the at least one identifier assigned to a source end system of the data packet, to forward the encapsulated packet to all or a subset of other switches in the network, and to send the original data packet to access ports having the at least one identifier; and  
20 upon receipt of the encapsulated packet at a receiving switch, de-encapsulate the packet and to forward the de-encapsulated packet to the access ports having the at least one identifier.

43. A computer-readable storage medium comprising program instructions for restricting flooding of a data packet, of one of a broadcast, multicast and unknown destination type, in a switch to be used in a switched data communications network, the network to include end systems and switches connected by links, the switches having access ports connected to end systems and network ports connected to other switches, the program instructions causing the switch to:

30 a. assign at least one identifier to a respective subset of end systems;

- b. map the at least one assigned identifier to an access port attached to at least one end system in the respective subset of end systems; and
- c. when the data packet is received from a source end system at a receiving access port of the switch:

- i) determine one or more identifiers associated with the source end system;
- ii) encapsulate the data packet by adding a header with the one or more determined identifiers;
- iii) forward the encapsulated data packet to all or a subset of other switches in the network; and
- iv) determine if at least one access port other than the receiving access port on the switch is associated with the one or more determined identifiers and forward the data packet out the at least one determined access port.

44. The computer-readable storage medium as recited in claim 43, further comprising instructions to cause the switch to:

- d. when an encapsulated data packet is received:
  - i) strip the header from the encapsulated data packet and determine the one or more encapsulated identifiers in the header of the encapsulated data packet;
  - ii) determine if at least one access port of the switch is associated with the one or more encapsulated identifiers; and
  - iii) forward the data packet out the at least one determined access port of the switch.

45. The computer-readable storage medium as recited in claim 43, further comprising instructions to cause the switch to, if in step c(iv) no other access port is determined, discard the data packet.

46. The computer-readable storage medium as recited in claim 43, further comprising, in step b, instructions to cause the switch to:

- maintain a first table to relate the at least one assigned identifier to the end systems or access ports of the switch; and
- maintain a second table to relate the access ports of the switch to assigned identifiers.

47. The computer-readable storage medium as recited in claim 46, further comprising, in step c.i), instructions to cause the switch to:  
review the first table for the one or more identifiers associated with the source end system or the receiving access port.

5 48. The computer-readable storage medium as recited in claim 47, further comprising, in step c.iv), instructions to cause the switch to:  
review the second table for an access port associated with the one or more determined identifiers.

10 49. The computer-readable storage medium as recited in claim 43, further comprising instructions to cause the switch to:  
prior to assigning an identifier to a specific end system or access port, maintain a default identifier for that specific end system or access port which maps to predetermined access ports.

15 50. The computer-readable storage medium as recited in claim 44, further comprising instructions to cause the switch to:  
listen to end systems heard on the access ports and to maintain the end systems heard and their respective access ports in a mapping table.

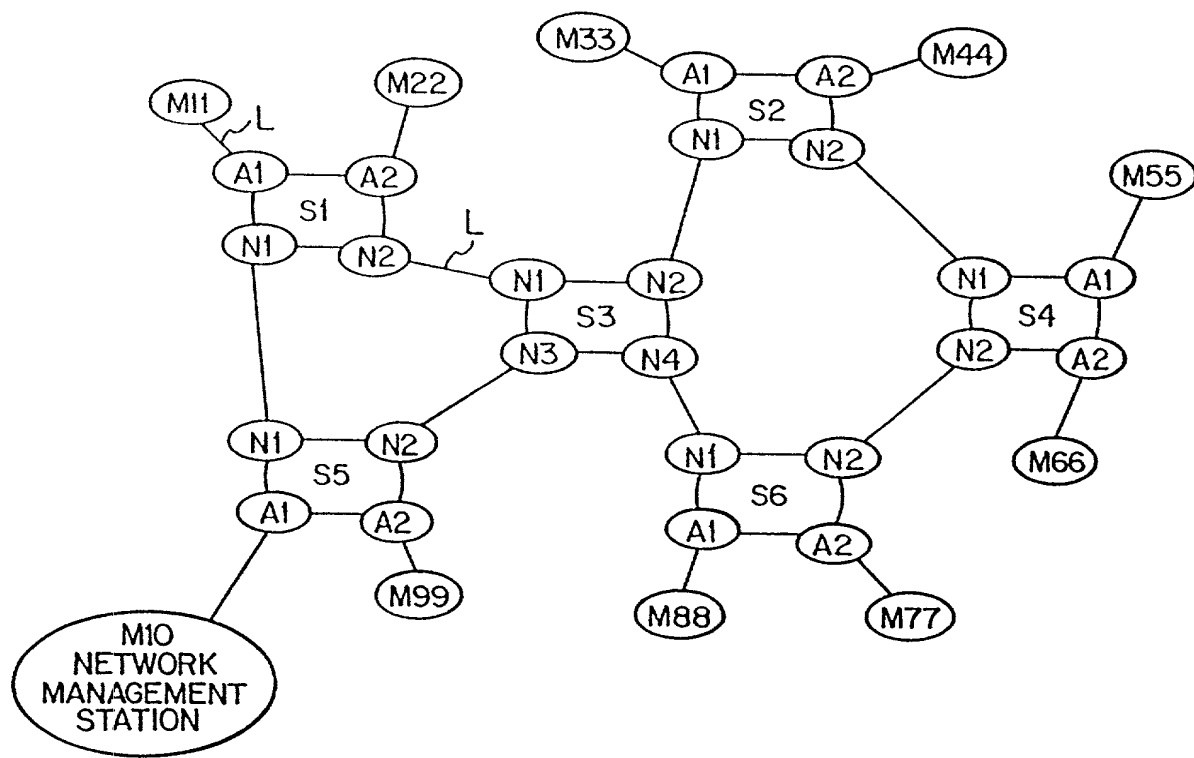
20 51. The computer-readable storage medium as recited in claim 43, further comprising instructions to cause the switch to:  
maintain a Management Information Base (MIB) interface.

52. The computer-readable storage medium as recited in claim 51, further comprising instructions to cause the switch to:  
use a Simple Network Management Protocol (SNMP) message to maintain a mapping table.

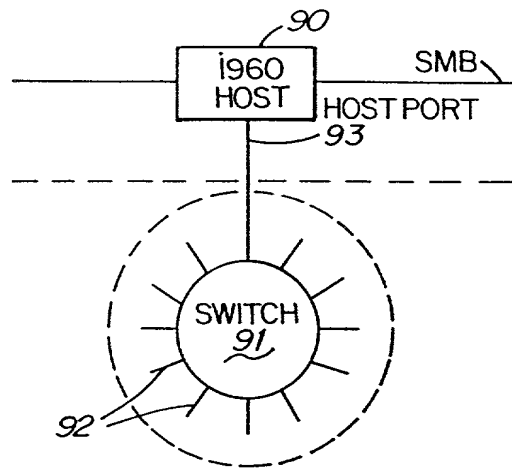
25 53. The computer-readable storage medium as recited in claim 43, further comprising instructions to cause the switch to:  
maintain a status table to enable and disable a respective subset.

5  
10

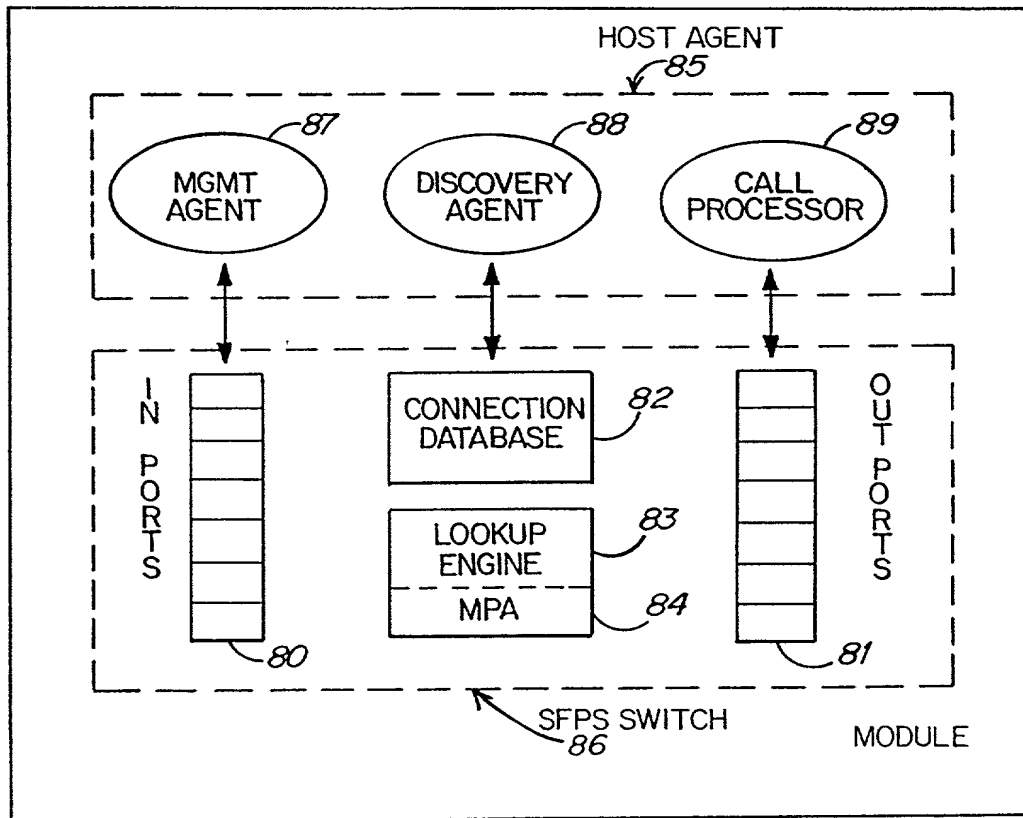
switched network. The method assigns different virtual LAN identifiers (VLAN-IDs) to different subsets of associated end systems or access ports. Tables are maintained for mapping the VLAN-IDs with associated end systems and access ports. When a broadcast packet is received at a first switch, it is encapsulated with a VLAN header, including the VLAN-IDs, and sent out a multicast channel to all other switches in the network (domain). The original packet is sent out the other access ports of the receiving switch for the designated VLAN-IDs. The switches receiving the VLAN packet remove the header and send the original packet out access ports associated with the VLAN-IDs extracted from the header. The method provides a mechanism for forwarding broadcast packets of a protocol not supported by the switching mechanism, as well as multicast packets and unicast packets from undiscovered end systems.



**FIG. 1**



**FIG. 2**



**FIG. 3**

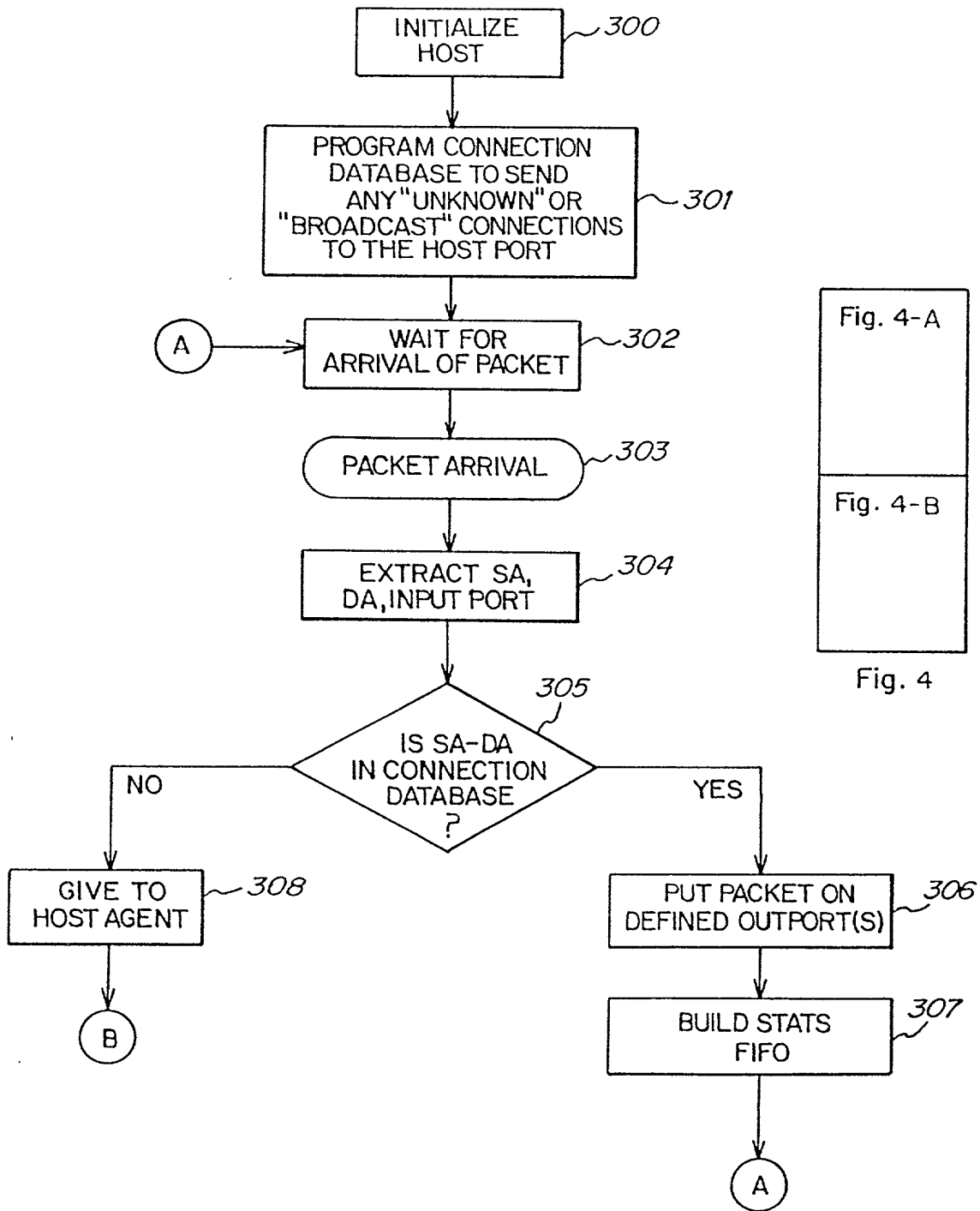


Fig. 4-A

Fig. 4-B

Fig. 4

FIG. 4-A



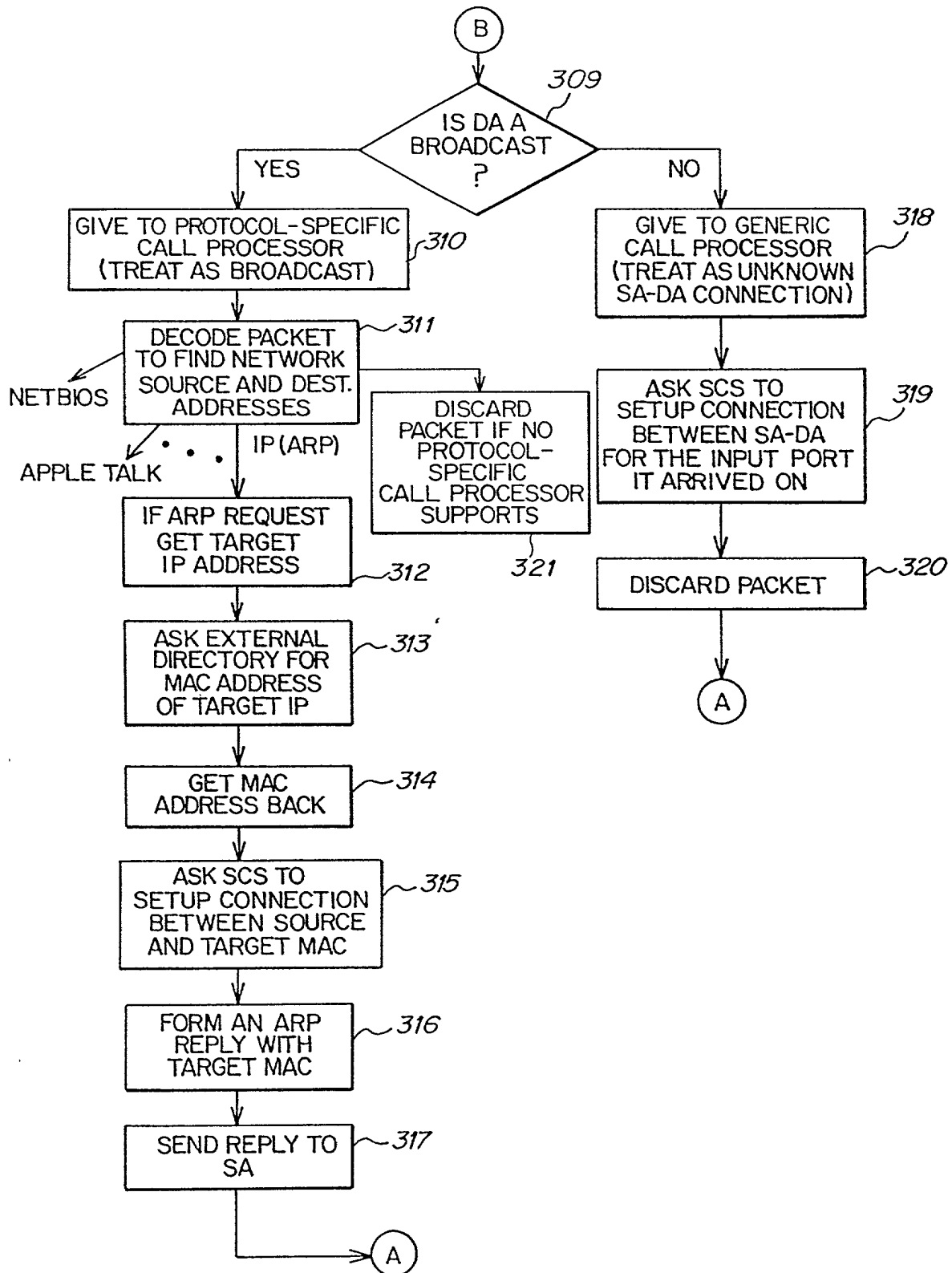
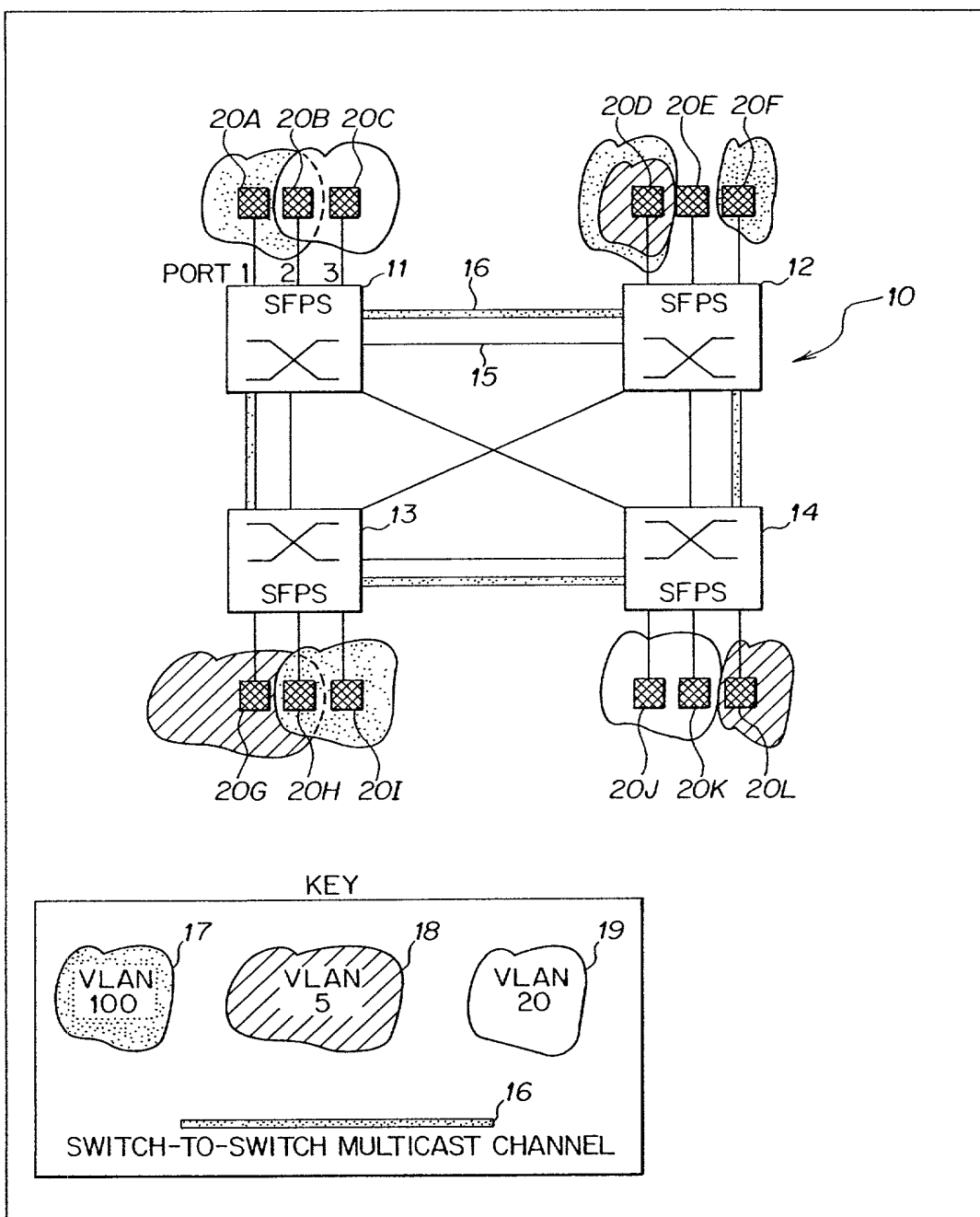
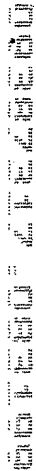
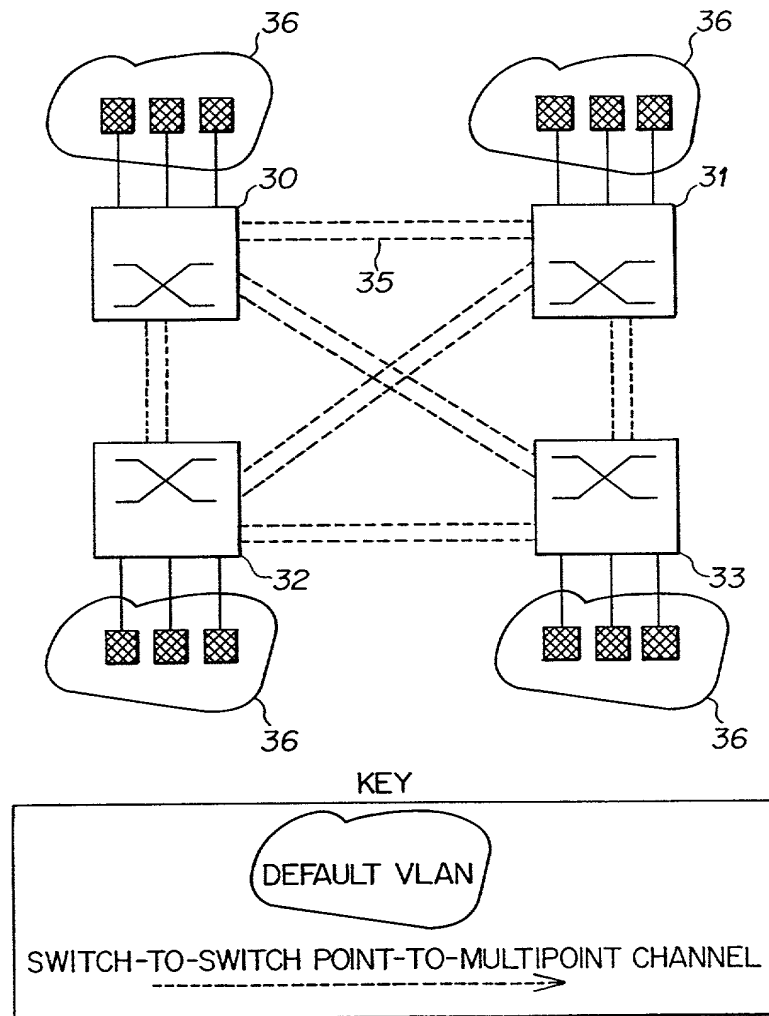


FIG. 4-B

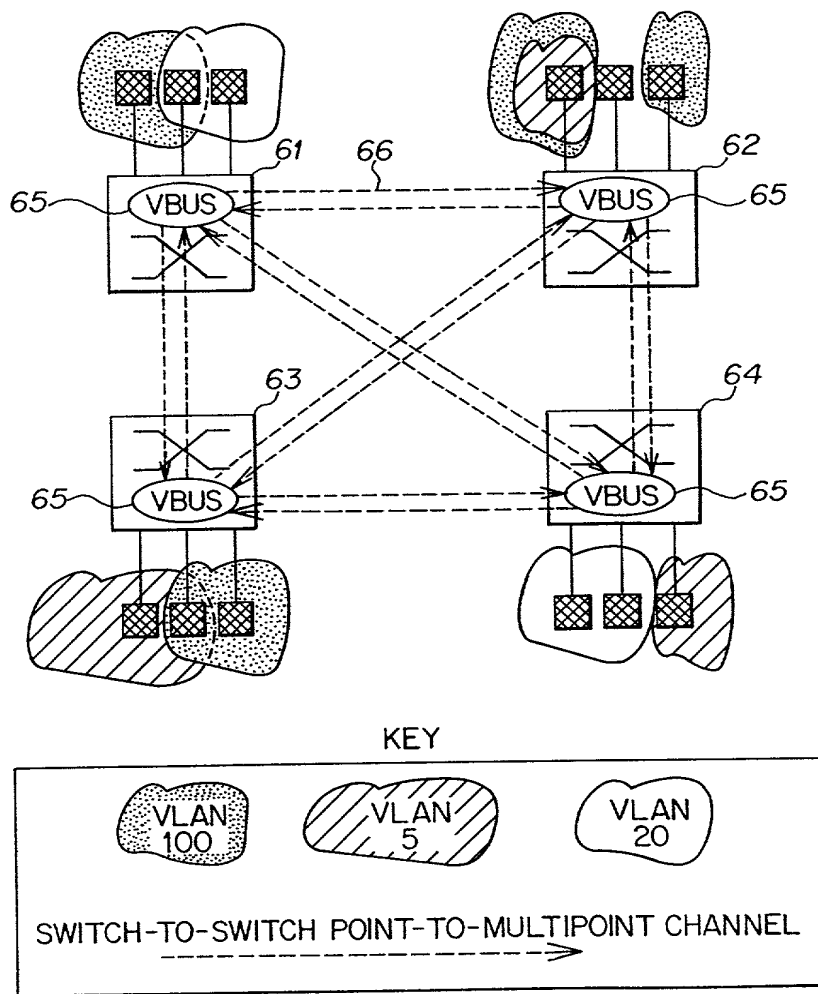


**FIG. 5**

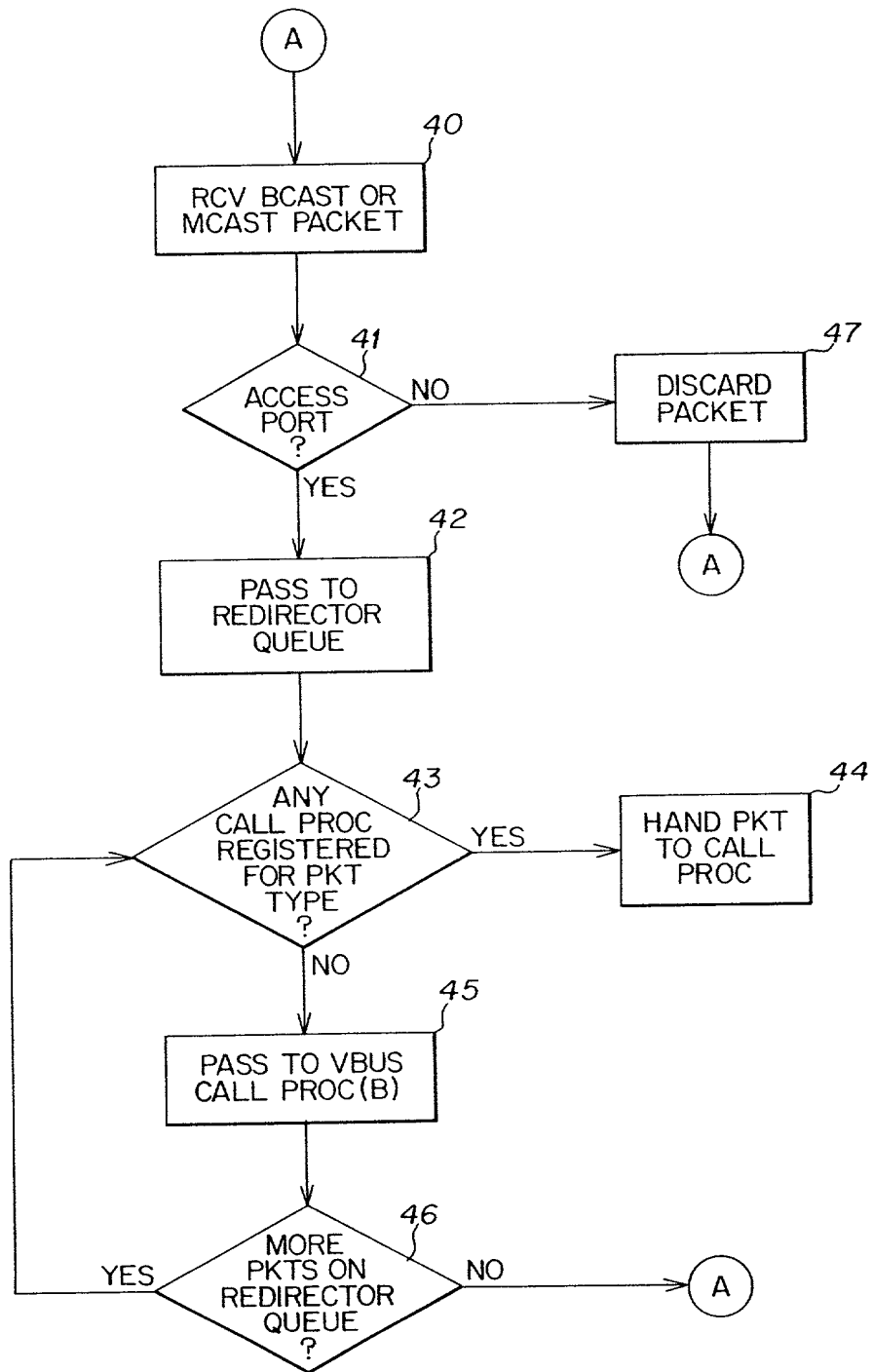
[illegible][illegible][illegible][illegible][illegible][illegible][illegible]



**FIG. 9**

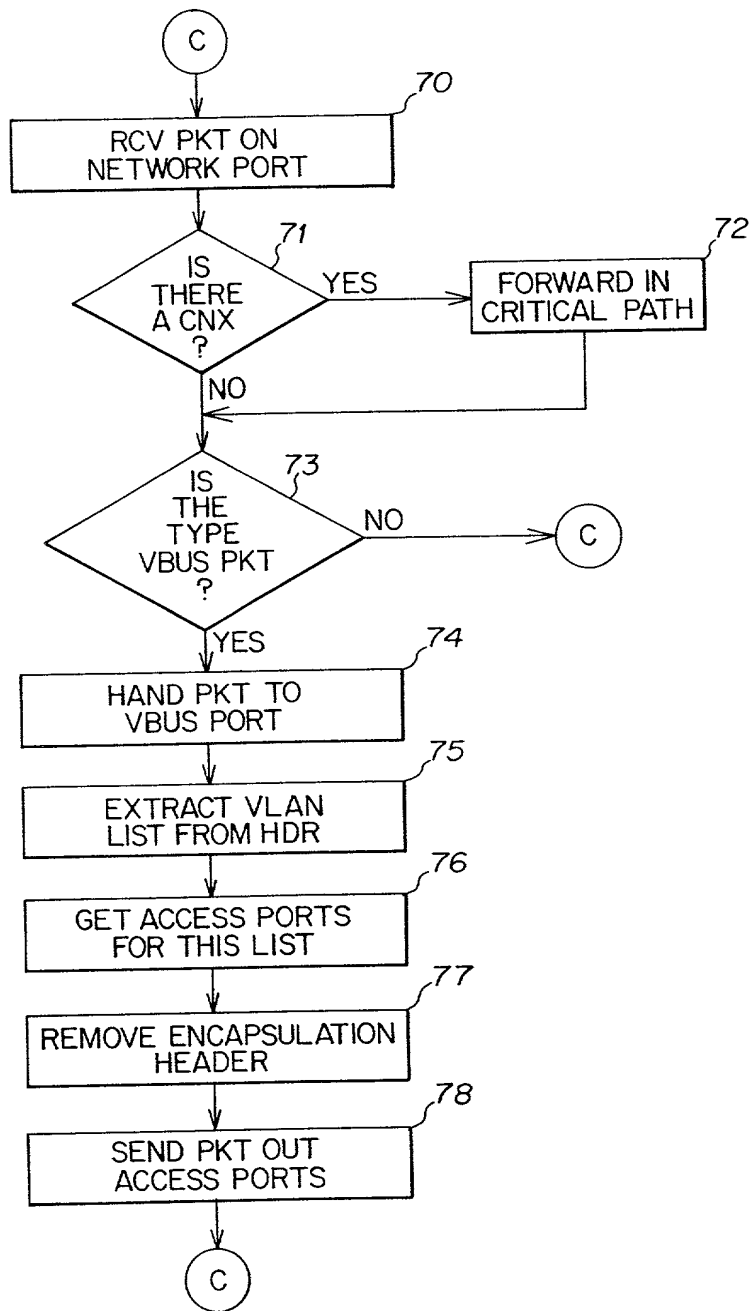


**FIG. 10**



**FIG. 11**





**FIG. 13**



## DECLARATION FOR PATENT APPLICATION

**COPY**

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name.

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

EXPRESS MAIL MAILING LABEL NO. EL056833040US

METHOD FOR ESTABLISHING RESTRICTED BROADCAST  
GROUPS IN A SWITCHED NETWORK

the specification of was filed on November 15, 1995 as Application Ser. No.  
08/559,738.

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, §1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign PCT Application(s) and any priority claims under 35 U.S.C. §119:			Priority Claimed	
(Number)	(Country if PCT so indicate)	(DD/MM/YY Filed)	[ ] YES	[ ] NO
(Number)	(Country)	(DD/MM/YY Filed)	[ ] YES	[ ] NO

I hereby claim the benefit under Title 35, United States Code, §119(e) of any United States provisional application(s) listed below.

(Application Number)	(filing date)
(Application Number)	(filing date)

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) or PCT international application(s) designating the United States of America listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56 which became available between the filing date of the prior application and the national or PCT international filing date of this application:

(Application Serial No.)	(filing date)	(status-patented, pending, abandoned)
(Application Serial No.)	(filing date)	(status-patented, pending, abandoned)

PCT Applications designating the United States:

(PCT Appl. No.) (U.S. Ser. No.) (PCT filing date) (status-patented, pending, abandoned)

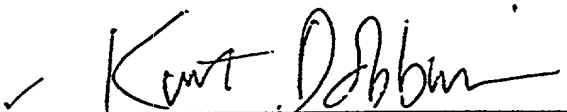
I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and to transact all business in the Patent and Trademark Office connected therewith:

Therese A. Hendricks  
M. Lawrence Oliverio

Address all telephone calls to Therese A. Hendricks at telephone no. (617) 720-3500.  
Address all correspondence to

Therese A. Hendricks  
c/o Wolf, Greenfield & Sacks, P.C.,  
Federal Reserve Plaza  
600 Atlantic Avenue  
Boston, MA 02210-2211

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.



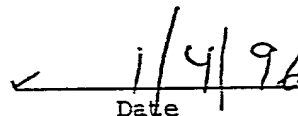
Inventor's signature

Full name of sole or first inventor: Kurt Dobbins

Citizenship: U.S.

Residence: 20 Harred Lane, Bedford, New Hampshire 03102

Post Office Address: Same

  
Date



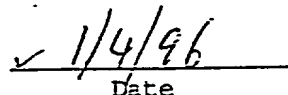
Inventor's signature

Full name of sole or first inventor: Phil Andlauer

Citizenship: U.S.

Residence: 253 Winding Pond Road, Londonderry, New Hampshire 03053

Post Office Address: Same

  
Date



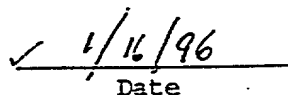
Inventor's signature

Full name of sole or first inventor: Michael Skubisz

Citizenship: U.S.

Residence: 1 Sandy Brook Drive, Durham, New Hampshire 03824

Post Office Address: Same

  
Date